



VICTORIA UNIVERSITY
MELBOURNE AUSTRALIA

Portable Device for Continuous Sensing with Rapidly Pulsed LEDs – Part 1: Rapid On-the-fly Processing of Large Data Streams using an Open Source Microcontroller with Field Programmable Gate Array

This is the Accepted version of the following publication

Noori, A, Mahbub, Parvez, Parry, JS, Davis, J, Lucieer, A and Macka, M (2019)
Portable Device for Continuous Sensing with Rapidly Pulsed LEDs – Part 1:
Rapid On-the-fly Processing of Large Data Streams using an Open Source
Microcontroller with Field Programmable Gate Array. Measurement. ISSN
0263-2241

The publisher's official version can be found at
<https://www.sciencedirect.com/science/article/pii/S0263224119304609>
Note that access to this version may require subscription.

Downloaded from VU Research Repository <https://vuir.vu.edu.au/38502/>

Accepted Manuscript

Portable Device for Continuous Sensing with Rapidly Pulsed LEDs – Part1:
Rapid On-the-fly Processing of Large Data Streams using an Open Source Microcontroller with Field Programmable Gate Array

Ansara Noori, Parvez Mahbub, John S. Parry, John Davis, Arko Lucieer, Mirek Macka

PII: S0263-2241(19)30460-9
DOI: <https://doi.org/10.1016/j.measurement.2019.05.034>
Reference: MEASUR 6643

To appear in: *Measurement*

Received Date: 7 January 2019
Revised Date: 26 April 2019
Accepted Date: 4 May 2019



Please cite this article as: A. Noori, P. Mahbub, J.S. Parry, J. Davis, A. Lucieer, M. Macka, Portable Device for Continuous Sensing with Rapidly Pulsed LEDs – Part1: Rapid On-the-fly Processing of Large Data Streams using an Open Source Microcontroller with Field Programmable Gate Array, *Measurement* (2019), doi: <https://doi.org/10.1016/j.measurement.2019.05.034>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.



Measurement

journal homepage: www.elsevier.com

Portable Device for Continuous Sensing with Rapidly Pulsed LEDs – Part 1: Rapid On-the-fly Processing of Large Data Streams using an Open Source Microcontroller with Field Programmable Gate Array

Ansara Noori^a, Parvez Mahbub^{a, b}, John S. Parry^c, John Davis^c, Arko Lucieer^d, Mirek Macka^{a, e, f*}

^a Australia Centre for Research on Separation Science (ACROSS) and School of Physical Sciences- Chemistry, University of Tasmania, Tasmania, Australia

^b Institute for Sustainable Industries and Livable Cities, Victoria University, Footscray Park Campus, Melbourne, Victoria 3011, Australia

^c Central Science Laboratory, University of Tasmania, Private Bag 74, Hobart 7001, Australia

^d School of Land and Food, University of Tasmania, Private Bag 76, Hobart 7001, Australia

^e Department of Chemistry and Biochemistry, Mendel University in Brno, Zemedelska 1, 613 00 Brno, Czech Republic

^f Central European Institute of Technology, Brno University of Technology, Purkynova 123, 612 00 Brno, Czech Republic. 118, 612 00, Brno, Czech Republic

ARTICLE INFO

ABSTRACT

Article history:

Received

Received in revised form

Accepted

Available online

Keywords:

Field programmable gate array

Red Pitaya microcontroller

On-the-fly data acquisition and processing

IR LED

Rapid microsecond pulsing

Portable sensor

We designed a portable system using an open source microcontroller (μC) with built-in field programmable gate array (FPGA) for on-the-fly data acquisition and processing of optical data generated from rapidly pulsed infrared light emitting diodes (IR LEDs) for optical sensing of gases. The system is used for rapid pulse generation (ca. 2 μs short pulses with a typical repetition rate of 1 kHz) to drive the IR LED, as well as for the optical sensing data acquisition and processing on-the-fly large data streams of ca. 2 Gbit/s. The flexibility and performance of the system is demonstrated. Each of the digitally processed signal pulses yielded one data point of analytical signal in time as a quasi-continuous data stream produced at a rate of between 1000 and 0.1 Hz. This microcontroller-based portable open source platform is then implemented in on-the-fly data acquisition and processing, of analytical signals enabling continuous gas sensing.

© 2017 Elsevier B.V. All rights reserved.

13 1. Introduction

14 For analytical measurements where the sample property can
 15 change at a fast rate, such as in the case of atmospheric
 16 monitoring of trace gases, rapid digital sampling and
 17 analysis techniques are required [1]. This requirement is
 18 well satisfied with optical analytical platforms, such as
 19 infrared (IR) spectrometers, supplemented with adequately
 20 fast electronics and data handling capabilities. Although IR
 21 spectroscopy-based gas detection is a well-established
 22 technique [2], designing small low-cost low power
 23 consumption analytical platforms for portable and remote
 24 analysis presents a number of challenges [3]. One of them is
 25 the rapid, on-the-fly processing of continuous and live data
 26 streams in a flexible custom-defined manner. Additionally,
 27 low-cost, small size and weight, and low-power analytical
 28 platforms capable of rapid, on-the-fly and custom-defined
 29 data processing are required in a number of field
 30 deployment modes including portable hand-held devices and
 31 remote sensing devices such as on-board unmanned aerial
 32 vehicles (UAVs).

33 Most gaseous analytes of environmental or industrial
 34 significance have strong absorption bands in the infrared
 35 (IR) spectral range [4]. Most commercially available
 36 instruments for the analysis of gases employ sophisticated
 37 and expensive spectrometers that provide measurements
 38 solely in a laboratory setting [5, 6]. Light emitting diodes
 39 (LEDs) have proven to be in many ways ideal light sources
 40 for optical detection and sensing in portable format [7-9]. In
 41 this context, the use of LEDs with photodiodes (PDs) in the
 42 IR spectral range has enabled the development of portable
 43 low-cost sensors [10-12]. Recently we demonstrated that
 44 response of MIR LED-based absorption photometric sensor
 45 for methane can be predicted from 1st principles using the
 46 readily available molecular absorption data HITRAN,
 47 resulting in calibration line slope agreement to $\pm 1\%$ with
 48 experimental data [13].

49 In this paper we advance this work in investigating a
 50 programmable fully portable sensor for methane using a
 51 powerful hand-held open source microcontroller with field
 52 programmable gate array (FPGA), to our best knowledge
 53 used for the first time for portable analytical
 54 instrumentation. The FPGA was capable of rapid on-the-fly
 55 processing of large data streams of up to 2GB/s which then
 56 resulted in the demonstration of continuous sensing of
 57 methane in indoor and outdoor environment (Part 2).

58 Most of the commercially available LED drivers for pulse
 59 signal generation (in μ s pulses with a 1kHz frequency) have
 60 fixed settings and need additional electronics for signal
 61 collection and data acquisition [14, 15]. In most cases an
 62 oscilloscope or standard electronic data acquisition (eDAQ)
 63 system can be a good option. However, the maximum
 64 sampling rate for data acquisition of a typical eDAQ is only
 65 1 kHz, which is not adequate for the acquisition of data
 66 generated by microsecond pulses [16]. Expensive digital
 67 oscilloscopes with sampling rates in excess of 100 mega
 68 samples/sec can be implemented for data acquisition and
 69 collection, however, the acquisition of large data streams (in
 70 our case 125 MHz/16bit yielding 2 GB/s) will exhaust the
 71 memory of a typical 16 GB SD card in only ca. 1 min [17].

72 Nowadays, computers are omnipresent as an interface with
 73 analytical instruments for online digital data acquisition,
 74 processing, storage, and display [18, 19]. Some are capable
 75 of precisely handling large data streams in modern

76 laboratory-based (not portable) analytical instruments such
 77 as Raman spectrometry [20]. However, for miniaturized
 78 portable analytical instruments, modern powerful
 79 microcontrollers (μ C) are ideal where on-the-fly (live data)
 80 data processing is needed [8, 21]. Recently the Hauser group
 81 published a review covering the use of μ C for portable
 82 analysis [8]. Although a number of μ C based commercial
 83 devices, including those for detection of methane, are
 84 available [22], these lack flexibility and give no insight into
 85 the way the analytical signal is produced ('black box'), so
 86 that it is in principle impossible to make a judgement on the
 87 data processing.

88 Currently, open source μ C systems, such as Arduino, are
 89 popular due to their programmable options [23], however,
 90 the Arduino can handle only one operation at a time and the
 91 maximum sampling rate of its in-built analog-to-digital
 92 converter (ADC) is only 10 kHz [24]. Another popular open
 93 source μ C, Raspberry pi, has no built-in analog input,
 94 therefore one has to be implemented using an additional
 95 ADC [25]. Conventional ADCs perform single conversions
 96 at a time, which results in a random lag between analog
 97 signal acquisition and data processing, making it difficult to
 98 generate synchronised data [26]. Importantly, even with a
 99 very fast ADC, the Raspberry pi is not capable of processing
 100 data in a rapid manner due to the speed limit of its
 101 processor. Regarding the most important parameter in
 102 respect to this work, namely on-the-fly large data stream
 103 data processing, both the Arduino and Raspberry pi would
 104 not be able to handle data streams in excess of 50 kHz at 16
 105 bit 0.1 MB/sec [27].

106 Alternatively, a recently introduced portable
 107 microcomputer with a field-programmable gated array
 108 (FPGA, which enhances the processing capabilities of
 109 existing microprocessors) 'Red Pitaya' (a technology spin-
 110 off from Instrument Technologies as the makers of Libera
 111 family devices [28]) is capable of on-the-fly processing of
 112 large data volumes without any lag, thanks to the FPGA
 113 responsible for data synchronization [29]. An FPGA allows
 114 for the integration of the ADC interface, input/output (I/O)
 115 interface, memory, and processing units in a single chip
 116 [30]. FPGA-based devices are especially used in particle
 117 colliders for high-energy physics (HEP) [31], gamma
 118 radiation spectroscopy, real vision imaging and many other
 119 types of reconfigurable high performance virtual
 120 instrumentation [32]. Although FPGA based devices offer
 121 real-time data handling capability of large data without any
 122 lag, other than in the fields of nanosecond pulse generation
 123 [33], computational chemistry [34] and simulated mass
 124 spectroscopy (MS) [35], the application of a μ C system with
 125 FPGA in analytical chemistry to the best of authors'
 126 knowledge, has not been presented in the analytical
 127 literature.

128 Therefore, we aimed to investigate flexible data
 129 acquisition and on-the-fly fully automated data processing
 130 through developing an in-house data processing routine
 131 capable of handling large and live data using a μ C system
 132 with an FPGA in a rapid manner. This creates the capability
 133 of generating rapid pulsed signals and processes in real time
 134 giving large amounts of data per second (2 Gbits/s at 125
 135 MHz with 16bit ADC), where implementation of a
 136 miniaturized μ C with an FPGA for IR LED based optical
 137 gas sensing offers portability, and at the same time
 138 maximum flexibility for implementing codes for specific

139 analytical scenarios such as portable and remote analysis of
140 gases.

141 2. Instruments and Methods

142 2.1. Instrumentation

143 2.1.1. Microcontroller with field programmable gate array

144 The microcontroller (μ C) system (Red Pitaya V1.1, RS
145 Components Pty Ltd, Wetherill Park, NSW, 1851,
146 Australia) shown in Fig. 1 is an open source platform, based
147 on an ARM Cortex A9 processor plus a Zynq μ C system on
148 chip (SoC) field programmable gate array (FPGA) in the
149 same device (component A in Fig. 1) with 512MB of DDR3
150 RAM (component B in Fig. 1). The operating μ C system is
151 based on Linux (version 2015.1 from Xilinx) supporting
152 network connection (WIFI, LAN and USB), which allows it
153 to operate remotely. The ARM CPU functions as a data
154 analyser to evaluate the data collected by a high-speed
155 ADC. The sampling capability of this μ C system through
156 RF output and input (components C and D in Fig. 1) has
157 four different options from 2-125 MHz. The buffer size
158 (maximum data capture capacity) of the FPGA- μ C system is
159 16,384 points. The input and output buffer of the FPGA- μ C
160 system was self-triggered using its external triggering
161 facilities in the GPIO (shown as component E in Fig. 1).

162 Insert figure 1

163 2.1.2. In-house electronics: Voltage to current converter 164 and resistor-capacitor circuit or RC filter.

165 We developed a voltage to current conversion unit (V-
166 to-I) and a resistor-capacitor circuit as an RC filter in-house
167 with off-the-shelf electronics. The V-to-I circuit converts a
168 voltage pulse, generated by the Red Pitaya FPGA, to a
169 current pulse to drive the LED.

170 The RC filter has two 1 ohm resistors and a 390 μ F
171 capacitor. There is also a dummy load resistor of 27 ohms
172 connected across the output. The dummy load is necessary
173 so that the power bank does not switch off if the load
174 becomes too small. The filter supplies power to the LED
175 driver (V-to-I) so that the 2 amp pulse is not affecting the
176 detector which is supplied from the same power bank. See
177 the supplementary information (SI) Fig. S1 A&B for a
178 detailed circuit diagram of the V-to-I, and RC filter.

179 2.1.3. LED and photodiode

180 We used an IR LED with an emission maximum
181 wavelength $\lambda_{\text{max}} = 1.65 \mu\text{m}$, (Lms16LED-R, Alfa Photonics,
182 Latvia) and an IR sensitive photodiode (PD) (Lms24-05-PA,
183 Alfa Photonics, Latvia) having a spectral response over the
184 range from 1.1 to 2.3 μm equipped with embedded
185 preamplifier.

186 2.1.4. Power supply.

187 We employed a rechargeable portable power bank
188 (CY1767PBCHE, Cygnett, Australia) to supply 5 volt DC
189 power to the μ C system, V-to-I conversion circuit, and the
190 preamplifier circuit of the IR PD.

191 2.2. Method

192 2.2.1. Pulse generation

193 We used the signal generation feature of the
194 microcontroller (μ C) system (Red Pitaya) to generate the
195 desired voltage pulses. Signal generation of the μ C operates
196 by filling a floating-point array of up to 16384 values with

197 the desired voltage at each time point, then commanding the
198 FPGA to produce that voltage pattern at the desired
199 frequency. They were subsequently converted into current
200 pulses by the voltage to current converter (V-to-I) unit to
201 drive the IR LEDs in pulse mode. The shape of the pulse is
202 flexible, generated stepwise digitally with details described
203 in the Results and Discussion section.

204 After receiving the radiation from the pulsed IR LED,
205 optical pulses from the IR PD were acquired by the μ C
206 system through Input Channel 1 as analog voltage pulses.
207 An 800 mm long and 7.5 mm inner diameter electro
208 polished aluminium tube was used as a sample cell for
209 portable analysis, which also housed the IR LED and the IR
210 PD (shown in Fig. 2 in the Results and Discussion section).

211 2.2.2. Fast data acquisition and on-the-fly data processing

212 The analog voltage pulses from the IR PD which were
213 collected by the μ C system through the RF input high speed
214 ADC (input channel 1) were designated 'raw pulses' for
215 clarity. We constructed a data processing program for the
216 ARM cortex A9 processor of the μ C system to perform
217 digital smoothing on the raw pulses. The digital smoothing
218 utilized three techniques: repetitive smoothing (averaging a
219 number of consecutive pulses) alone, and in addition to the
220 repetitive smoothing, boxcar averaging, or Savitzky-Golay
221 smoothing (a special form of 2nd polynomial regression-
222 based smoothing). After the application of the smoothing
223 techniques, the smoothed pulses were termed 'processed
224 pulses' (for more information see Fig. 4).

225 The baseline and pulse top of each of the processed
226 pulses were evaluated using three different statistical
227 operations: averaging, linear regression, and 2nd degree
228 polynomial regression. From the evaluated values of
229 baseline and pulse-top, the height of the pulse was
230 calculated (by subtraction), resulting in one data point for
231 each processed pulse, termed the 'final signal value'(S).
232 After acquiring additional quasi-continuous final signal
233 values from an arbitrary number of pulses, the 'digital data
234 signal' in volts was formulated and by taking the negative
235 natural logarithm (base e) of the data stream values, the
236 digital data stream was converted into the 'final analytical
237 signal' (A) in absorbance units (A.U.) (for more information
238 see Fig. 4). The total calculation time for the μ C was
239 ~20 ns.

240 Baseline noise was evaluated by observing the
241 distribution pattern of all the data points in the baseline of
242 each of the pulses for random and fixed instrumental noise
243 in our detection system. The distribution pattern for the final
244 digital signal was tested on the obtained results using two
245 different statistical evaluation techniques: simple averaging,
246 and linear regression. Both the instrumental and analytical
247 signal to noise ratio was calculated and the result was
248 optimized by comparing one with another.

249 3. Results and discussions

250 3.1. Design of Pulse Generation for IR-LED

251 The radiometric power output of LEDs increases
252 proportionally with the magnitude of the applied current
253 [36]. However, the temperature across the chip of the LED
254 rises significantly when it is driven at a higher applied
255 current [37], which causes efficiency droop (i.e. the
256 efficiency of the LED decreases while operated with higher
3

257 electric current) due to overheating across the
258 semiconductor material of the LED chip [36].

259 Therefore, to minimize the effect of overheating of the
260 semiconductor materials used in the IR LED, the LEDs have
261 to be operated either in a quasi-continuous wave (QCW)
262 mode (duty cycle = 50%) or in pulsed mode (switched on
263 for a very short time, usually microseconds). The maximum
264 driving current in QCW for the IR LED is 250 mA [38],
265 whereas in pulse mode the driving current can be up to 2A
266 [38], which yields higher radiometric power output during
267 the pulse [36] and this in turn yields in better performance
268 of the optical measurement due to lower minimum
269 absorbance values that can be measured by absorbance-
270 based analytical detection [15, 39].

271 In rapid pulsing mode, the duty cycle (the percentage of
272 the 'on' time) is significantly shorter, which helps to reduce
273 the thermal effect. In our work, the IR LED was in "on"
274 mode only for 2 μ s with a duty cycle equal to 0.2%, so the
275 LED was in "off" mode for a comparatively longer period
276 (998 μ s), which allows sufficient time to cool down and
277 protect the LED from efficiency droop. The corresponding
278 pulse repetition frequency (PRF) in our study was 1 kHz,
279 which helps to produce a higher number of pulses within a
280 short period of time with resulting maximum radiometric
281 power output. Hence, 0.2% duty cycle provides data
282 processing suitability at such high PRF since the pulse width
283 determined the number of data points to be processed.

284 To further demonstrate the flexibility of this approach
285 with custom-defined data processing, we developed a
286 computer program written in C and compiled in a Linux OS
287 environment to generate the voltage pulses in the required
288 shape. This program was employed to forward an array of
289 voltage values, with stepwise amplitudes between 0 and
290 1 volt, from the μ C system to the voltage-to-current
291 converter (V-to-I) circuit shown in Fig. 2. Since LEDs are
292 current driven, we applied the in-house voltage-to-current
293 (V-to-I) converter circuit to transform the voltage pulsed
294 signal generated (V_{in}) by the μ C system into current signal
295 pulse (I_m) for the IR LED.

296 Insert Figure 2

297 As mentioned before, generated pulses can be produced
298 in arbitrary shapes i.e. any time duration (pulse width or
299 duty cycle), frequency, and forward voltage are possible
300 simply by changing the parameters of the program. The
301 stepwise pulses generated by the μ C are shown in Fig. 3A
302 (i) as a continuous stream and in Fig. 3B (i) as single pulse
303 which was constructed by the following: 500 steps with 0
304 volts to achieve the base line, 20 steps to achieve 0.9 volt,
305 180 steps to make the pulse top with 0.9 volt, 20 steps to
306 bring the pulse signal down to 0 volt and the remaining
307 steps to fill the buffer at 0 volt (1 step = 10ns. The pulse
308 generated was repeated with 1 kHz frequency, and the total
309 time duration depends on the number of pulse data that need
310 to be processed. The corresponding converted currents
311 pulsed from the V-to-I conversion unit, and measured in
312 channel 2 of the μ C are shown in Fig. 3 A (ii) and 3 B (ii).
313 These currents were used to drive the IR LEDs in pulse
314 mode. IR radiation from the LED was detected by the IR
315 PD and transformed from an optical pulse signal to voltage
316 pulses (V_{out}) as measured in Channel 1 of the μ C shown in
317 Fig. 3A (iii) and 3B (iii). This voltage pulsed signal was
318 collected by the μ C system as a raw pulsed signal and
319 employed for further processing.

320 Insert Figure 3

321 From Fig. 3 B i) it is observed that the stepwise
322 generated pulse from the μ C system follows a smooth
323 shape, with a sharp rise and fall as it is generated. However,
324 when it was converted into current pulses by the V-to-I
325 conversion circuit the LED has a rise time of 200 ns to
326 generate the final optical output (~ 2 A). After detecting the
327 response from the IR LED, the IR PD has a "rise time" and
328 a "fall time" of 250 ns as shown in Fig. 3 B. The response
329 delay appears due to inherent properties of the
330 semiconductor material of the IR LED and IR PD and the
331 response of the embedded PA and therefore, cannot be
332 controlled by the user.

333 3.2. Data Acquisition and on-the-fly Data processing with 334 μ C system

335 By default, Red Pitaya FPGA performs data
336 acquisition in continuous mode, which may result in
337 overwriting and loss of necessary data for further processing
338 in pulsed mode. Therefore, data acquisition was performed
339 through a command to the FPGA to acquire a full buffer of
340 16384 points as an array of digital numbers. Triggering was
341 used to ensure the μ C system only collected the informative
342 part of the raw signal that included the baseline and the
343 entire pulse.

344 To eliminate time lag between data acquisition and
345 data processing, the input and output buffer of the μ C were
346 synchronized using self-triggering. Self-triggering was
347 performed by employing the external triggering facility of
348 the FPGA where the digital output from the GPIO
349 (component E in Fig. 1) was fed back as the external trigger.
350 It was programmed by raising the pin from low (0 volt) to
351 high (3.3 volt), keeping the pin high for 5 μ s then
352 allowing it to fall back to low, triggering both RF input and
353 output to function simultaneously

354 3.2.1. Digital filtering by repetitive smoothing, boxcar 355 averaging and Savitzky-Golay smoothing

356 In order to achieve smooth pulses from the
357 acquired raw pulses, we incorporated three digital filtering
358 techniques namely, repetitive smoothing, boxcar averaging,
359 and Savitzky-Golay smoothing through C programming in
360 the CPU of the μ C. When digital filtering software is
361 incorporated in commercially available analytical
362 instruments, analysts lose the flexibility of investigating
363 different digital filtering techniques with variable input data
364 according to specific analytical requirements [15]. Our
365 digital filtering approach with the μ C system allowed us to
366 select flexible numbers of raw pulses starting from time zero
367 (triggered on) to perform the repetitive smoothing by
368 averaging consecutive pulses [40]. Then boxcar averaging,
369 and Savitzky-Golay (S-G) methods [15] using point wise
370 data after the repetitive smoothing also produced smoothed
371 processed pulses. The selection criteria for the large number
372 of raw pulses for processing are described in the SI.

373 In the data processing program, we started with
374 repetitive smoothing of different numbers of raw pulses
375 being averaged ($A = 10, 100, 1000$, and $10,000$) (Shown in
376 Fig. 4A). We then employed Boxcar averaging and
377 Savitzky-Golay methods on 1000 pulses already smoothed
378 by repetitive smoothing to investigate whether further
379 smoothing after repetitive smoothing is required (discussed
380 in section 3.3.2).

381 In order to obtain the final signal (height of the pulses)
382 from both raw and processed pulses we applied three

383 different statistical operations in the program: simple
384 averaging, linear regression, and 2nd degree polynomial
385 regression, to evaluate the base line and pulse top (shown in
386 Fig. 4B). At 0.2% duty cycle with 100M/s data rate, 200
387 points were delivered while the LED is on only for 2 μ s.
388 Therefore, we selected 125 data points (discarding the pulse
389 rise and fall) from the pulse top and 150 data points from the
390 base line (before the rise) for each statistical operation. The
391 difference between the pulse top and baseline i.e. pulse
392 height is considered the final signal value for each
393 individual pulse. Schematic representations of the final
394 evaluated signal as pulse height and the final stream of
395 quasi-continuous data are shown in Fig. 4C.
396 Insert Figure 4

397 3.2.2. Baseline noise evaluation and instrumental signal-to- 398 noise (SNR) from the processed pulses

399 A. Baseline noise evaluation.

400 Theoretically, Gaussian (white) noise attenuates with
401 square root of the number of repetitive pulses, while other
402 types of noise will not, and therefore the additional Boxcar
403 and Savitzky-Golay smoothing along with the repetitive
404 smoothing might not be equally beneficial. To assess the
405 nature of the noise of the processed data, we have conducted
406 statistical analysis based on histograms of the baseline data
407 point values. The histograms were constructed using 150
408 data points ($A=1000$) in the baseline of processed pulses,
409 using repetitive smoothing, Boxcar and repetitive
410 smoothing, and Savitzky-Golay and repetitive smoothing as
411 shown in Fig. 5A. From the histograms in Fig. 5A we
412 observed that the baseline signals follow a normal
413 distribution for repetitive smoothing and for repetitive
414 smoothing + Boxcar, while the repetitive smoothing +
415 Savitzky-Golay resulted in a distribution skewed to the
416 right. The characteristic appearance of the normal
417 distribution of the baseline data values in this study
418 confirms that the baseline signals resulting from repetitive
419 smoothing and repetitive smoothing + Boxcar averaging
420 methods include only white noise [41, 42].

421 We determined the baseline noise of the processed
422 pulses after repetitive smoothing by multiplying the
423 standard deviation (σ) of 150 baseline data points by 5,
424 following a classical noise evaluation technique in analytical
425 chemistry for flow-through detection, and found a good
426 agreement with theoretical noise values (as shown in SI Fig.
427 S2A). This also confirms the Gaussian nature of the baseline
428 noise in the processed pulses through repetitive smoothing.
429 The baseline noise values after applying the three different
430 smoothing techniques are shown in SI Fig. S2B. As
431 expected, the repetitive smoothing followed by additional
432 smoothing techniques resulted in lower white noise in the
433 baseline with Boxcar averaging being 7% and Savitzky-
434 Golay 5% lower.

435 The flexibility in this type of baseline noise evaluation
436 with the μ C system provides users with the capability to
437 choose from a variety of digital filtering-by-smoothing
438 techniques. Boxcar averaging and Savitzky-Golay
439 smoothing didn't result in a statistically significant
440 reduction in noise, so we have chosen repetitive smoothing
441 only for further investigation in this study with a view to
442 providing rapid data processing with the proposed μ C based
443 detection system.

444 3.2.3. Instrumental signal-to-noise (SNR)

445 The quality of an analytical method is very often quantified
446 by analyzing the signal-to-noise ratio (SNR). We
447 determined the instrumental SNR for each resulting signal
448 value (pulse height) and baseline noise obtained from each
449 processed pulse after employing different smoothing
450 techniques. In Fig. 5B we compare the SNR values
451 obtained after employing repetitive smoothing as a function
452 of the number of pulses being averaged (A) using three
453 different statistical methods. We observed that the SNR
454 improves as the number of pulses being averaged (A)
455 increases. The enhancement of SNR follows the theory
456 where SNR improves linearly by a factor of \sqrt{A} (shown in
457 Fig. 5C) [42]. However, different statistical operations have
458 no observable effect on the SNR values. All the
459 measurements were reproducible since the standard
460 deviations were too small to notice the error bars for 10
461 repetitions of each result.

462 Insert Figure 5

463 3.2.4. Determination of the digital data stream

464 A. Pulse top and baseline signal distribution

465 In order to select the most suitable statistical method for
466 the determination of the pulse top and baseline signal values
467 for each subsequent pulse signal (pulse height), we
468 investigated the distribution of the pulse top and baseline
469 signal values, applying simple averaging and linear
470 regression for both cases. We omitted the 2nd degree
471 polynomial as we did not observe any significant difference
472 in the SNR values using simple averaging, linear intercept,
473 or 2nd degree polynomial methods as illustrated in Fig. 5B.
474 In Fig. S3A of the SI we have shown the baseline and pulse
475 top signal values obtained from evaluating 10,000
476 consecutive raw pulses (without applying any digital
477 smoothing) using simple averaging and linear regression.
478 We constructed the histogram using these data values
479 (10,000 baselines and pulse tops) as shown in Fig. S3B and
480 S3C, and we observed that the simple averaging method
481 resulted in normal distributions as well as smaller standard
482 deviations when compared to linear regression for both
483 evaluations of baselines and pulse tops.

484 B. Final quasi-continuous data stream.

485 From the distribution pattern of baseline and pulse top
486 signal values (Fig. S3B & S3C) it is evident that simple
487 averaging has less deviation when it is used for signal
488 evaluation. Therefore, we considered the simple averaging
489 method to investigate the final signal (pulse height)
490 distribution while averaging consecutively increased
491 numbers of pulses for repetitive smoothing, and referred to
492 the result as the final digital data stream as shown in Fig. S4.
493 The distributions of the final pulsed signal (height of the
494 pulse) became smooth and, as expected, the analytical noise
495 (in voltage) of the final data stream lowered as the number
496 of consecutive pulses being averaged for repetitive
497 smoothing increased.

498 C. Evaluation of analytical signal by converting the 499 voltage signal into an absorbance signal.

500 Since the ultimate usage of the proposed optical system
501 is analytical sample detection based on the absorbance
502 principle, we converted each voltage signal of the final data
503 stream (i.e., Fig. S4) into an absorbance unit (A.U.). By
504 taking the negative natural logarithm (base e) of the final
505 signal (pulse height) values, the minimum measurable

absorbance was determined [41] and referred to as the final analytical signal for this proposed absorbance based detection system. In Fig. 6 the final absorbance signal is shown with corresponding analytical noise values (ΔA in A.U.). It is evident that as the number of repetitions of pulses for smoothing increases, the noise drops by a factor of the square root of the number of pulses, which consequently helps to improve the performance of any analytical detection. However, while the number of repetitive pulses increases, the time required for processing each pulse increases simultaneously and the rapid instantaneous processing capability of the system therefore decreases. Hence, where fast data processing is the principal focus, the number of repetitive pulses being smoothed needs to be optimized.

Insert Figure 6

3.3. Optimization.

In this section, the number of pulses to be averaged for repetitive smoothing are optimized for the proposed IR detection system using on-the-fly data processing which will be exercised further in the field for real sample analysis. For this, the analytical signal-to-noise ratios (ratios of the averaged absorbance values obtained from Fig. 6 to the corresponding absorbance noise values ($A/\Delta A$)) were compared with the instrumental signal-to-noise ratio (SNR) shown in Fig. 7A. In both cases, the signal-to-noise ratio increased as the number of pulses being averaged increased and almost the same pattern was followed although they were appraised independently for different data sets following different evaluation approaches. The analytical or absorbance noise values (ΔA) were also compared with the instrumental or baseline noise values (N) of each processed pulse as shown in Fig. 7B. From Fig. 7B, we observe that the analytical and instrumental noise values merge with each other at $A=10,000$ (for repetitive smoothing) which is the consequence of fixed instrumental noise and the pulse being almost smoothed. Although $A=10,000$ gives the optimal result in terms of noise elimination and signal-to-noise ratio enhancement, the time needed for each pulse to be smoothed for $A=10,000$ is 10 seconds, which in some cases may not be ideal in terms of rapid data processing. Therefore, to keep the system response fast enough for most real-time sensing scenarios, $A=1000$ (1 second for each data point) was chosen for on-the-fly data processing for in-field real sample analysis.

Insert Figure 7

4. CONCLUSIONS

This study demonstrates the prospects for rapid data processing of large data streams on modern portable μC systems with field-programmable gate arrays, for on-the-fly and rapidly changing sample scenarios. Further it shows the benefits of flexibility and full insight based on a custom data handling routine implemented in an open source μC system with a FPGA. The user-defined data processing thus acquired and implemented through the μC system in a flexible manner also allows the generation of the required pulsed signal with any desired shape, duration, frequency, and amplitude to drive the LED. The user can define and adapt the data processing software and apply it in a flexible way as required. The use of such miniaturized μC -FPGA systems with custom data processing routines and high

reproducibility makes the analytical sensing of rapidly changing samples, such as atmospheric gases, a real and relatively low-cost possibility.

Acknowledgments

MM acknowledges his ARC Future Fellowship Level 3 (FT120100559). The authors have declared no conflict of interest.

Supplementary information available

Detail of the in-house made instruments and figure for baseline noises, Pulse top and baseline signal values, constructed histograms and quasi-continuous digital data stream using two different statistical methods and command line parameter for program are given in the SI.

References

- 580 [1] J. Leis, D. Buttsworth, C. Snook, G. Holmes, Detection of
582 potentially explosive methane levels using a solid-state infrared
583 source. *IEEE Trans. Instrum. Meas.*, 63 (2014) 3088-3095.
- 584 [2] J. M. Dang, L. Fu, Z. H. Yan, C. T. Zheng, Y. C. Chang, C.
585 Chen, Y. D. Wang, A Review of Mixed Gas Detection System
586 Based on Infrared Spectroscopic Technique *Spectrosc. Spectral*
587 *Anal.* 34 (2014) 2851-2857.
- 588 [3] S. Fanchenko, A. Baranov, A. Savkin, V. Sleptsov, LED-
589 based NDIR natural gas analyzer. In *IOP Conference Series:*
590 *Materials Science and Engineering.* 108 (1) IOP Publishing,
591 2016. [http://iopscience.iop.org/article/10.1088/1757-
592 899X/108/1/012036/pdf](http://iopscience.iop.org/article/10.1088/1757-592/899X/108/1/012036/pdf)
- 593 [4] L. S. Rothman, I. E. Gordon, Y. Babikov, Y., Barbe, A.,
594 Benner, D. C., Bernath, P. F., et al (2013). The HITRAN2012
595 molecular spectroscopic database. *J. Quant. Spectrosc. Radiat.*
596 *Transfer*, 130 (2013) 4-50.
- 597 [5] K.-H. Kim, S. K. Pandey, R. Pal, Analytical bias among
598 different gas chromatographic approaches using standard BTX
599 gases and exhaust samples, *J. Sep. Sci.* 32 (2009) 549-558.
- 600 [6] N. S. Lawrence, Analytical detection methodologies for
601 methane and related hydrocarbons. *Talanta* 69 (2006) 385-392.
- 602 [7] M. Macka, T. Piasecki, P. K. Dasgupta, Light-emitting
603 diodes for analytical chemistry, *Annu. Rev. Anal. Chem.* 7
604 (2014) 183-207.
- 605 [8] D. A. Bui, P. C. Hauser, Analytical devices based on light-
606 emitting diodes—a review of the state-of-the-art. *Anal. Chim.*
607 *Acta.* 853 (2015) 46-58.
- 608 [9] A. Noori, P. Mahbub, M. Dvořák, A. Lucieer, M. Macka,
609 Radiometric analysis of UV to near infrared LEDs for optical
610 sensing and radiometric measurements in photochemical
611 systems, *Sens. Actuators, B Chem.* 262(2018)171-179.
- 612 [10] C. Massie, G. Stewart, G. McGregor, J. R. Gilchrist,
613 Design of a portable optical sensor for methane gas detection.
614 *Sens. Actuators, B*, 113 (2006) 830-836.
- 615 [11] G.-T. Park, K.-C. Park, G.-J. Lyu, J.-R. Kwon, Y.-G.
616 Kim, B.-J. Ryou, J.-I. Park, In 24th World Gas Conference in
617 October 2009: Buenos Aires, Argentine Republic, 2009.
- 618 [12] K. M. G. de Lima, A portable photometer based on LED
619 for the determination of aromatic hydrocarbons in
620 water. *Microchem. J.* 103 (2012) 62-67.
- 621 [13] P. Mahbub, J. Leis, and M. Macka, Chemometric approach
622 to the calibration of light emitting diode based optical gas
623 sensors using high-resolution transmission molecular
624 absorption data, *Anal. Chem.* 90(2018) 5973-5976.
- 625 [14] D-51i Universal Led Driver Instruction Manual,
626 http://lmsnt.com/datasheets/Electronics/D51i_en-230317.pdf (:
627 13 November, 2018
- 628 [15] LED Pulse Drivers, [http://www.roithner-
629 laser.com/led_pulsed_driver.html](http://www.roithner-629 laser.com/led_pulsed_driver.html), (11 November, 2018)
- 630 [16] EDAQ data recording made simple,
631 <https://www.edaq.com/>, (accessed: 13 November, 2018)
- 632 [17] Digital Oscilloscope, www.tek.com/digital-oscilloscope,
633 (12 November, 2018)
- 634 [18] T. A. O'Haver, Pragmatic Introduction to Signal
635 Processing, Univ. of Maryland at College Park, College Park,
636 MD, 1997, 58–69.
- 637 [19] L. F. Capitán-Vallvey, N. Lopez-Ruiz, A. Martinez-
638 Olmos, M. M. Erenas, A. Palma, Recent developments in
639 computer vision-based analytical chemistry: A tutorial review,
640 *J. Anal. Chim. Acta* 899 (2015) 23-56.
- 641 [20] D. Lauwers, P. Brondeel, L. Moens, P. Vandenabeele, In
642 situ Raman mapping of art objects. *Phil. Trans. R. Soc. A*, 374
643 (2016) 20160039
- 644 [21] Y. Suzuki, T. Takahashi, T. Takayanagi, S. Motomizu, S.
645 Kawakubo, Simple Kits of Colorimeter and Potentiometer for
646 High School and Undergraduate Student Education, *Bunseki*
647 *Kagaku* 59 (2010) 125-130.
- 648 [22] TMG: solutions by industry,
649 <http://www.tmgtestequipment.com.au/products/>, (accessed: 28
650 October, 2018)
- 651 [23] I. J. Koenka, J. Sáiz, P. C. Hauser, Instrumentino: An
652 open-source modular Python framework for controlling
653 Arduino based experimental instruments. *Comput. Phys.*
654 *Commun.* 185 (2014) 2724-2729
- 655 [24] P. Teikari, R. P. Najjar, H. Malkki, K. Knoblauch, D.
656 Dumortier, C. Gronfier, H. M. Cooper, An inexpensive
657 Arduino-based LED stimulator system for vision research. *J.*
658 *Neurosci. Methods* 211 (2012) 227-236.
- 659 [25] V. Gandhi, S. Heda, R. Anand, A. Zarin, A. Upadhyay, A.
660 L. Chakraborty, A Raspberry Pi-based field-deployable tunable
661 diode laser spectroscopy system for the detection of CO₂ at
662 2003.5 nm. In *Microwave and Photonics (ICMAP), 2015*
663 *International Conference on; IEEE, 2015*, pp 1-2.
664 <https://ieeexplore.ieee.org/abstract/document/7408713/>
- 665 [26] M. Abdallah, O. Elkeelany, A. Alouani, A conceptual
666 design of a compact multi-channel real-time analog signal
667 acquisition and processing system. In *41st Southeastern*
668 *Symposium on System Theory, 2009*, pp 360-362.
- 669 [27] S. Scholl, The Xilinx Zynq: A Modern System on Chip for
670 Software Defined Radios, (2016). In [https://kluedo.ub.uni-
671 kl.de/frontdoor/index/index/docId/4442](https://kluedo.ub.uni-671 kl.de/frontdoor/index/index/docId/4442), (1 August, 2018)
- 672 [28] A. Kosicek, Libera electron beam position processor, In
673 *Particle Accelerator Conference, 2005. PAC 2005. Proceedings*
674 *of the; IEEE, 2005*, pp 4284-4286.
- 675 [29] Redpitaya, Stemplab, <https://redpitaya.com/>, (7 November,
676 2018)
- 677 [30] H. Zhu, L. Sun, Z. Wang, Z. Nie, H. Liu, Design of
678 infrared methane gas concentration detection system, *J. App.*
679 *Opt.* 35 (2014) 890-894.
- 680 [31] K. T. Pozniak, FPGA-based, specialized trigger and data
681 acquisition systems for high-energy physics experiments *Meas.*
682 *Sci. and Technol.* 21 (2010) 062002.
- 683 [32] J. D. D. Gazzano, M. L. Crespo, A. Cicuttin, F. R. Calle,
684 Field-Programmable Gate Array (FPGA) Technologies for
685 High Performance Instrumentation; IGI Global, 2016.
- 686 [33] Y. Zhu, Wang, L. M. Design and Implementation of
687 Nanosecond Pulse Generator based on Reconfiguration PLL in
688 FPGA, In *Proceedings of the 2015 International Conference on*
689 *Electronic Science and Automation Control*, Liu, C., Ed.;
690 Atlantis Press: Paris, 2015, pp 323-326.
- 691 [34] A. Berces, B. Feher, P. Szanto, I. Pechan, L. Lajko, Z.
692 Runyo, P. Laczko, J. Lazanyi, FPGA implementation of
693 cheminformatics and computational chemistry algorithms and

694 its cost/performance comparison with GPGPU, cloud
695 computing and SIMD implementations Abstracts of Papers of
696 the American Chemical Society 240 (2010) 1.

697 [35] C. Pascoe, D. Box, H. Lam, A. George, FPGA-Accelerated
698 Isotope Pattern Calculator for Use in Simulated Mass
699 Spectrometry Peptide and Protein Chemistry, IEEE. In 2012
700 Symposium on Application Accelerators in High Performance
701 Computing; IEEE Computer Soc: Los Alamitos, 2012, pp 111-
702 120.

703 [36] Energy, U. S. D. O., Ed.; Office of Energy Efficiency &
704 Renew-able Energy: Washington, DC 20585, 2013.

705 [37] J. V. Lawler, J. Currano, Thermal simulations of packaged
706 IR LED arrays, In Technologies for Synthetic Environments:
707 Hardware-in-the-Loop Testing XIII: Orlando, FL, 2008.

708 [38] Alfa Photonics Ltd., <http://www.alfaphotonics.lv/>, (1
709 November, 2018)

710 [39] X. H. Geng, D. P. Wu, Q. Wu, Y. F. Guan, A compact and
711 highly sensitive light-emitting diode-induced fluorescence
712 detector for capillary flow systems, Talanta 100 (2012) 27-31.

713 [40] M. F. Wahab, P. K. Dasgupta, A. F. Kadjo, D. W.
714 Armstrong, Sampling frequency, response times and embedded
715 signal filtration in fast, high efficiency liquid chromatography:
716 A tutorial Anal. Chim. Acta 907 (2016) 31-44.

717 [41] J. M. Palmer, B. G. Grant, The Art of Radiometry,
718 Bellingham: SPIE Press, 2010.

719 [42] P. D. Wentzell, C. D. Brown, Signal Processing in
720 Analytical Chemistry, In Encyclopedia of Analytical
721 Chemistry, Meyers, R. A., Ed.; John Wiley & Sons Ltd,
722 Chichester, 2000, pp 9764-9800.

723 **Figures**

724

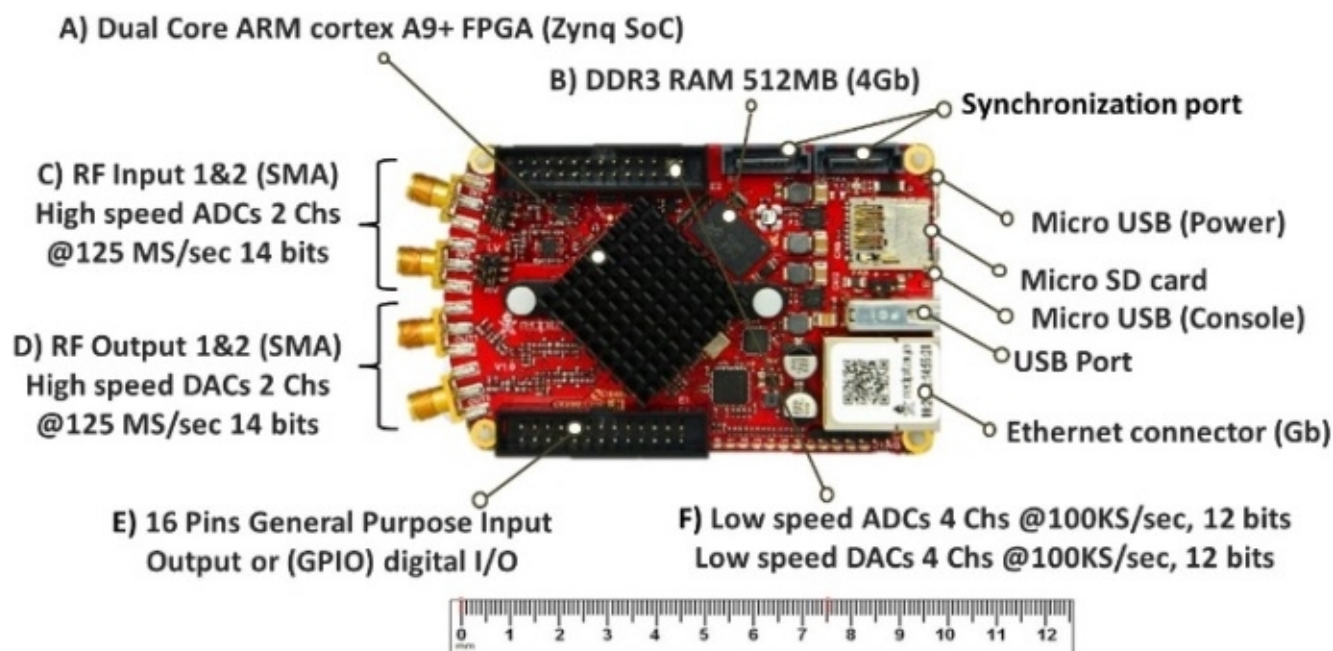


Fig. 1. μ C system (Red Pitaya) showing the principal components: A) μ C system Processor + FPGA, B) RAM High speed & resolution ADC input, C) High speed and resolution ADC input, D) High speed and resolution DAC output, E) General Purpose input and output (GPIO) which provides external self-triggering facilities, F) High speed ADC input and DAC output.

729

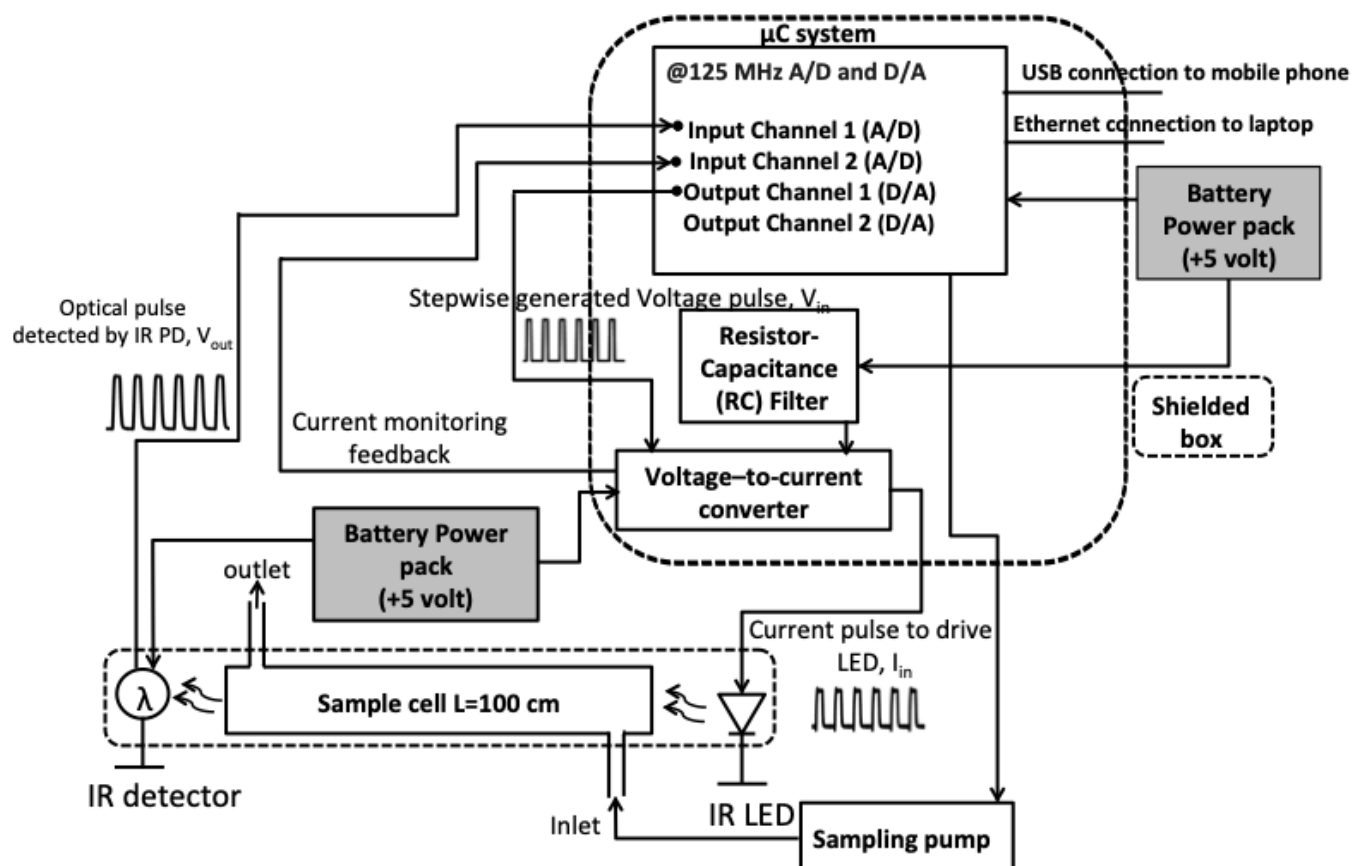


Fig. 2. Schematic representation of the IR detection associated with μ C system as pulse generation and data collection system.

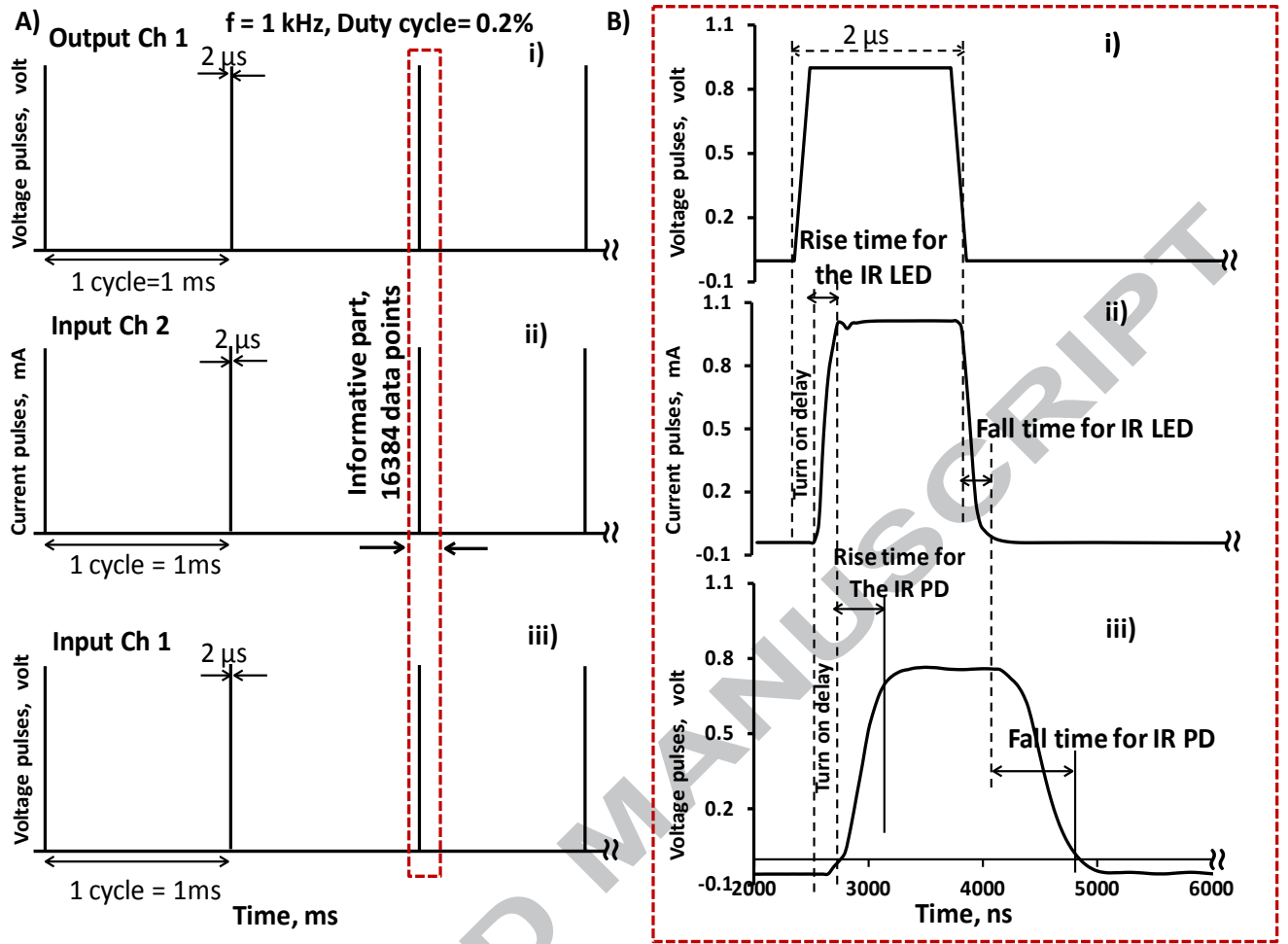


Fig. 3 A) Schematic representation of three types of pulsed signals generated and measured through i) output channel 1, ii) input channel 2 and iii) input channel 1 of the micro-controller system, informative parts (buffer size data points) are shown within the red dash lined rectangle. B) detail of each pulsed signal: (i) Step wise generated pulse defined by the μ C system (Red Pitaya); (ii) current pulse to drive the IR LED, and (iii) the corresponding optical output pulses from the IR LED detected by the IR PD.

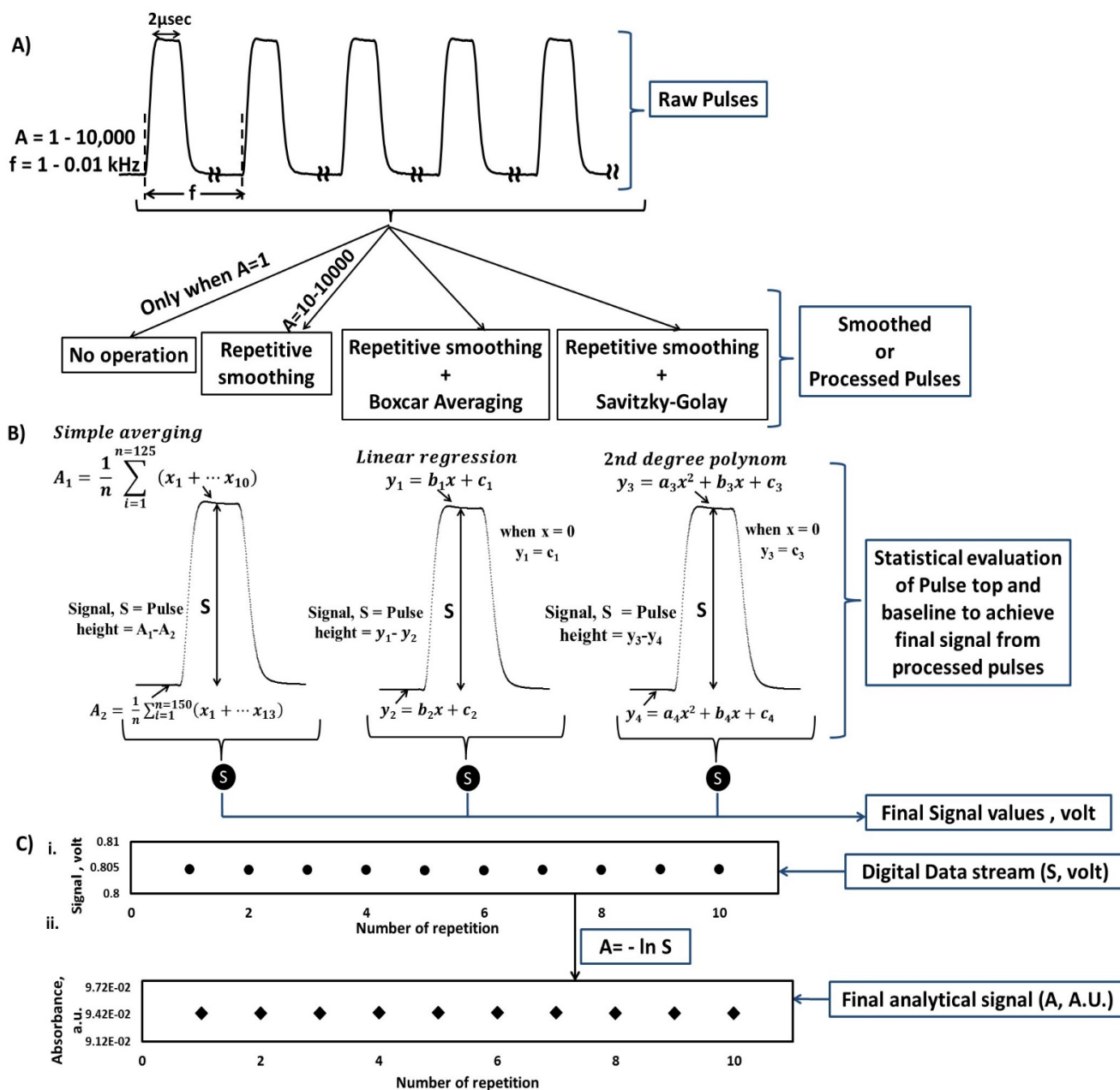


Fig. 4 Schematic representation of data processing methods A) repetitive smoothing and two additional digital smoothing techniques: Boxcar averaging and Savitzky-Golay applying on pointwise obtained data from repetitive smoothed pulses B) evaluation of baseline (from 150 data points) and pulse top (from 125 data points) of each processed pulse applying three different statistical methods to obtain final signal values (height of the pulses, S) C) Final data i. digital data stream in volt and ii. analytical signal in absorbance units (A.U.) from each individual pulse after the statistical evaluation of pulse top and baseline.

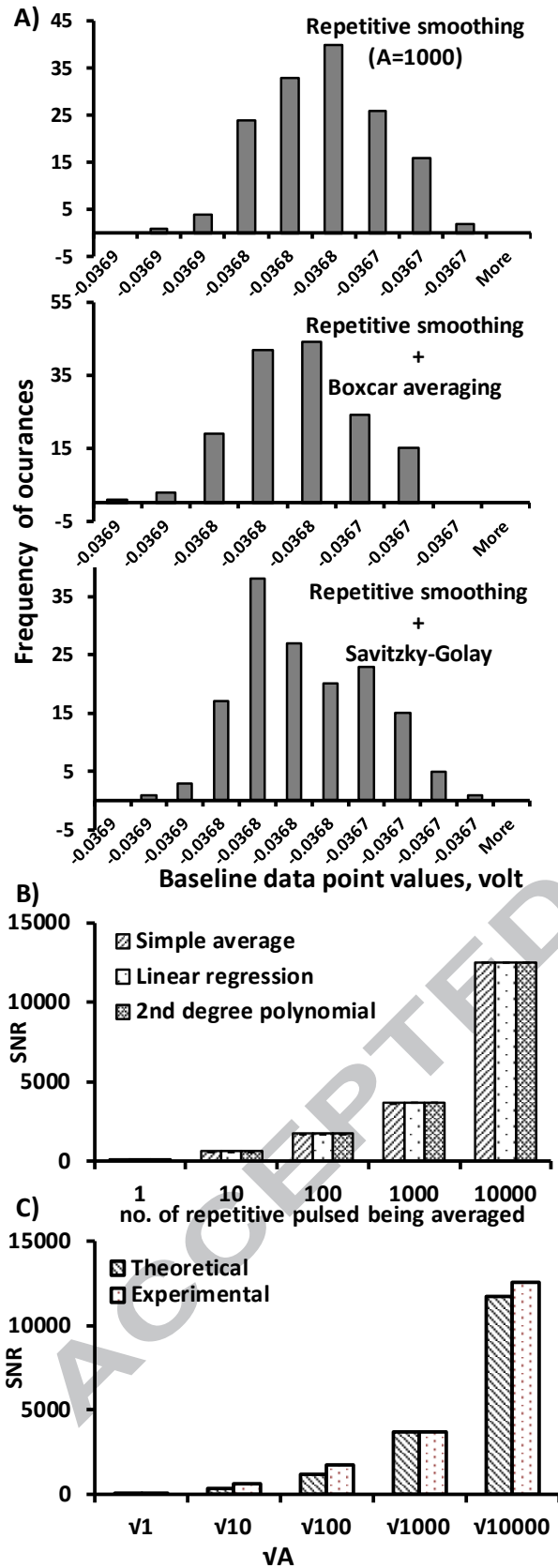


Fig. 5 A) Histograms of the baseline data points. The value of the baseline data points was obtained by binning the baseline data values, the difference between each of the bin values was $= (\text{max baseline} - \text{min baseline})/6$. The histogram covers the whole range of baseline data point values. The number of points averaged (A) was 1000. B) Signal-to noise ratio obtained by applying three different statistical methods after smoothing raw pulses by repetitive smoothing and C) comparison of experimental SNR with theoretical values where SNR should increase by a factor of \sqrt{A} .

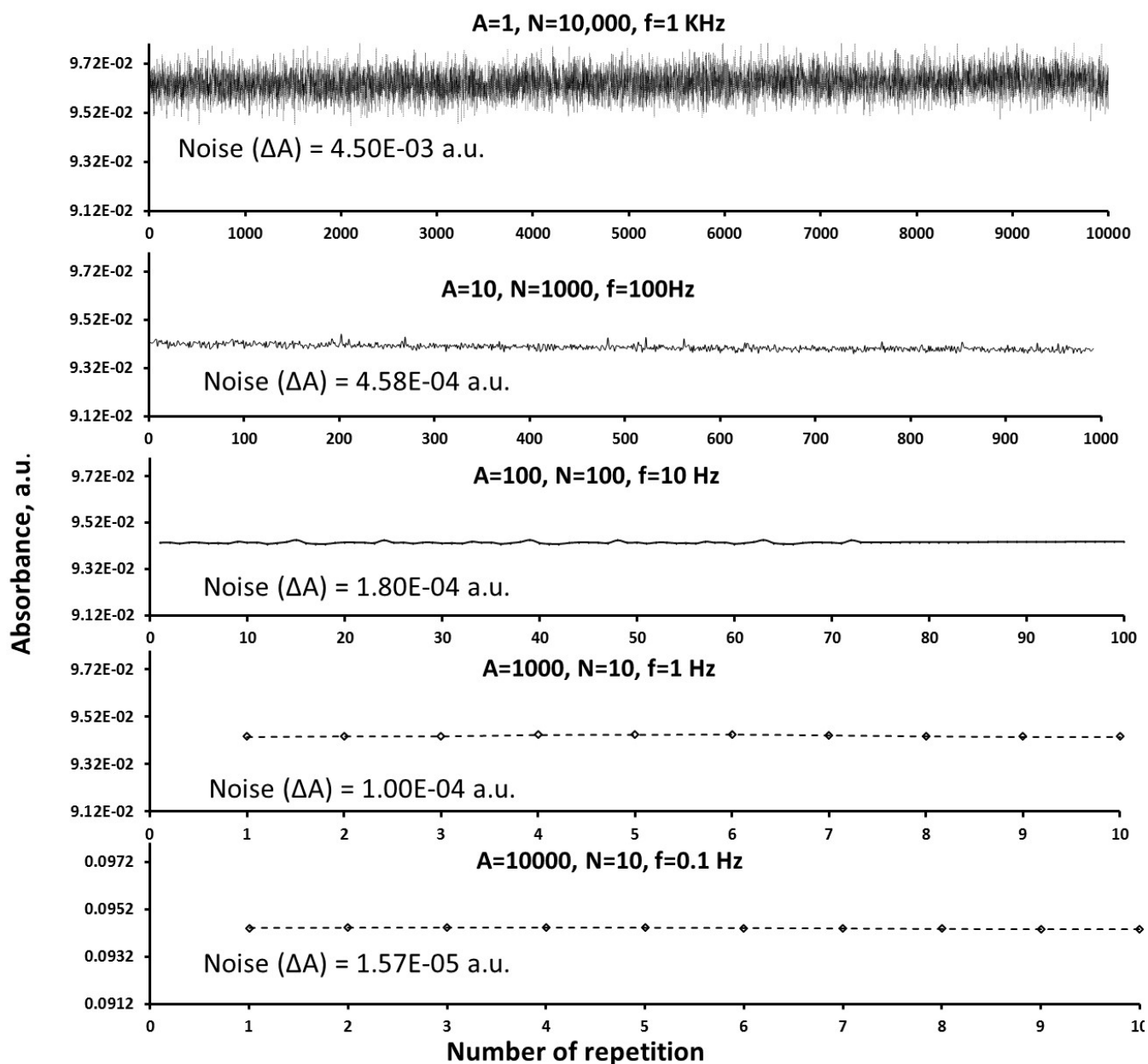


Fig. 6. Ultimate analytical signal in absorbance unit (A.U.) as negative natural logarithm of each signal values with corresponding noise values.

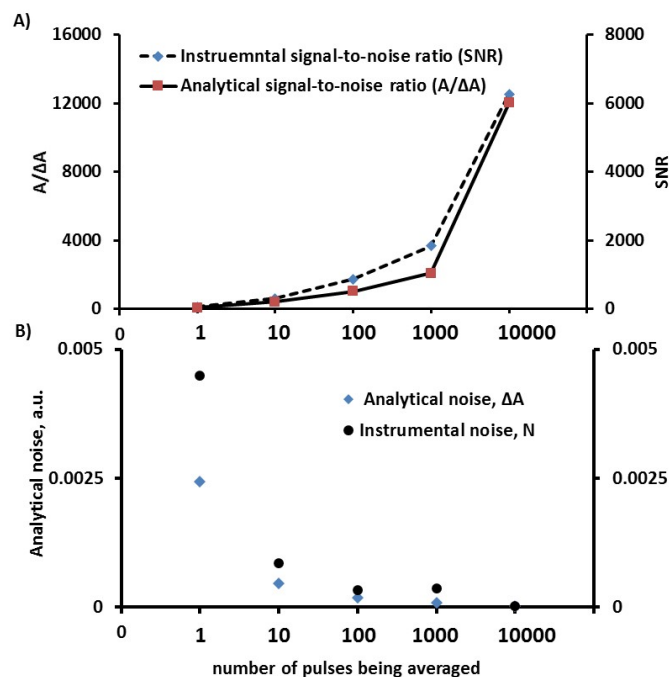
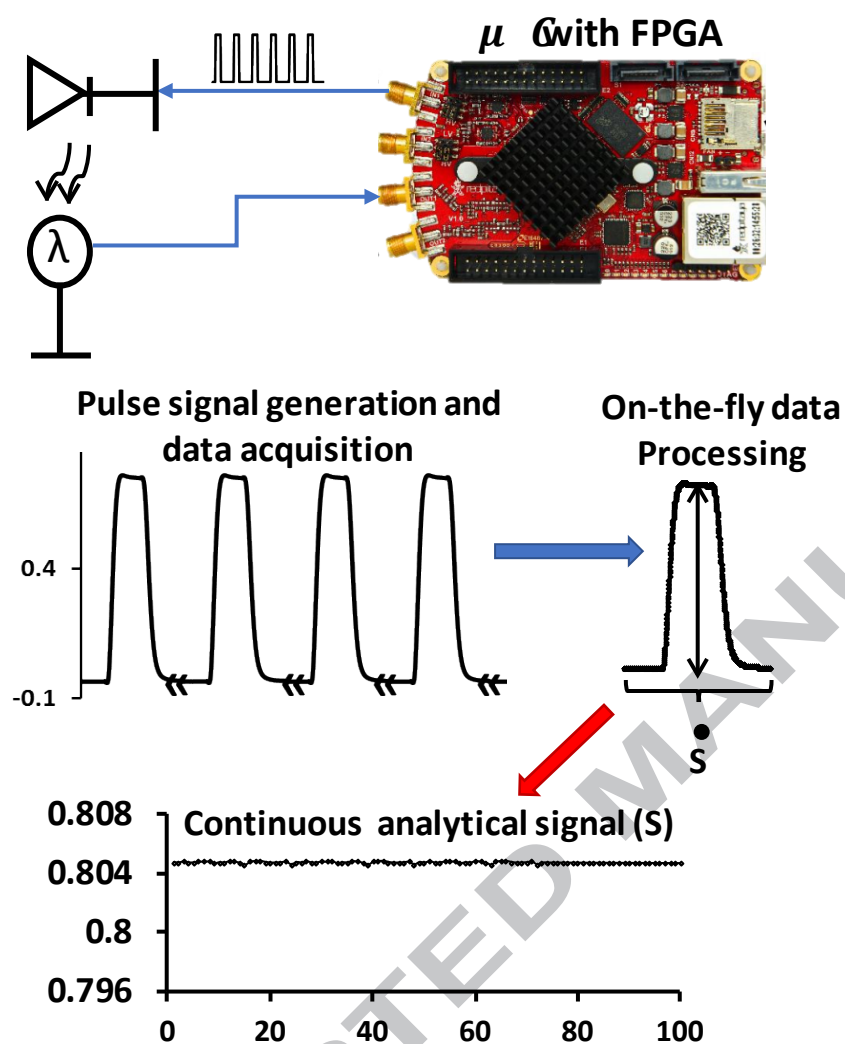


Fig. 7. Comparison of analytical and instrumental A) signal-to-noise ratio ($A/\Delta A$) and (SNR) respectively, calculated for different set of acquired data at different start time. B) noises ($\text{noise} = 5 \cdot \sigma$) of same set data as A.

GRAPHICAL ABSTRACT ONLY



Supplementary Information

Portable Device for Continuous Sensing with Rapidly Pulsed LEDs – Part 1: Rapid On-the-fly Processing of Large Data Streams using an Open Source Microcontroller with Field Programmable Gate Array

Ansara Noori^a, Parvez Mahbub^{a, b}, John S. Parry^c, John Davis^c, Arko Lucieer^d, Mirek Macka^{a, e, f*}

^a Australia Centre for Research on Separation Science (ACROSS) and School of Physical Sciences- Chemistry, University of Tasmania, Tasmania, Australia

^b Institute for Sustainable Industries and Livable Cities, Victoria University, Footscray Park Campus, Melbourne, Victoria 3011, Australia

^c Central Science Laboratory, University of Tasmania, Private Bag 74, Hobart 7001, Australia

^d School of Land and Food, University of Tasmania, Tasmania, Australia

^e Department of Chemistry and Biochemistry, Mendel University in Brno, Zemedelska 1, 613 00 Brno, Czech Republic

^f Central European Institute of Technology, Brno University of Technology, Purkynova 123, 612 00 Brno, Czech Republic.

1. Voltage-to-current conversion

The amplified output voltage from pin 6 of the op amp turns on the Mosfet (Fig. S1 A Q1) allowing current to flow through the LED (Fig. S1 D2) and resistors R10, R11. The voltage increased across the resistors until it is equal to the voltage applied to pin 3. This voltage is fed back to the op amp (D) through pin 2 which holds the current constant until the voltage applied to pin 3 changes to a different value

2. Other Figures

Figures for baseline noises, pulse top and baseline signal values, constructed histograms and quasi-continuous digital data stream using two different statistical methods are given in Figure S2, S3, S4 respectively. In Figure S2 A the theoretical noise was calculated using equation 1,

$$N_{theoretical} = \frac{S_{max} - S_{min}}{6} \dots \dots \dots (1)$$

Where, $N_{theoretical}$ is the base line noise, S_{max} is the maximum signal value in the baseline and S_{min} is the minimum signal value in the baseline.

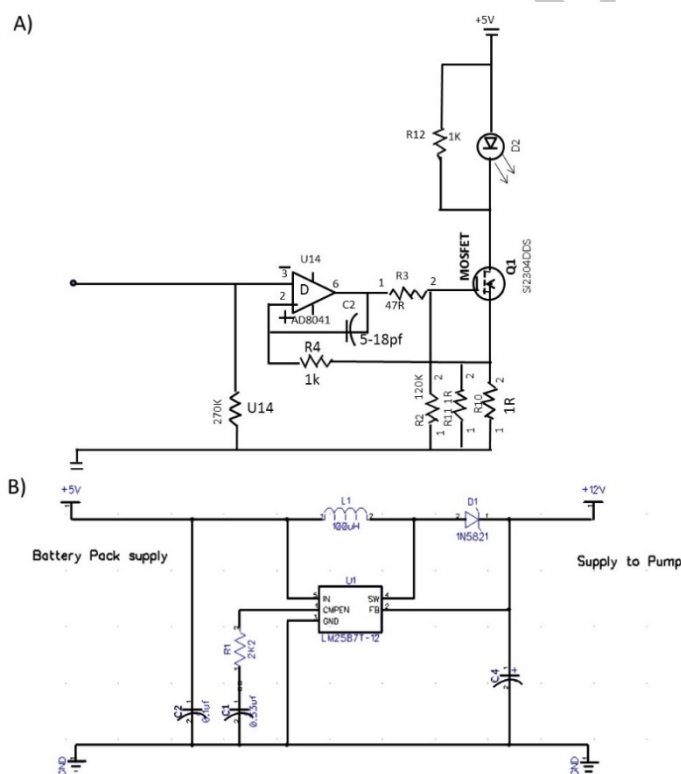


Fig. S 1 Block diagram of the in-house made A) voltage-to-current (V-to-I) conversion circuit B) resistor-capacitor (RC) filter.

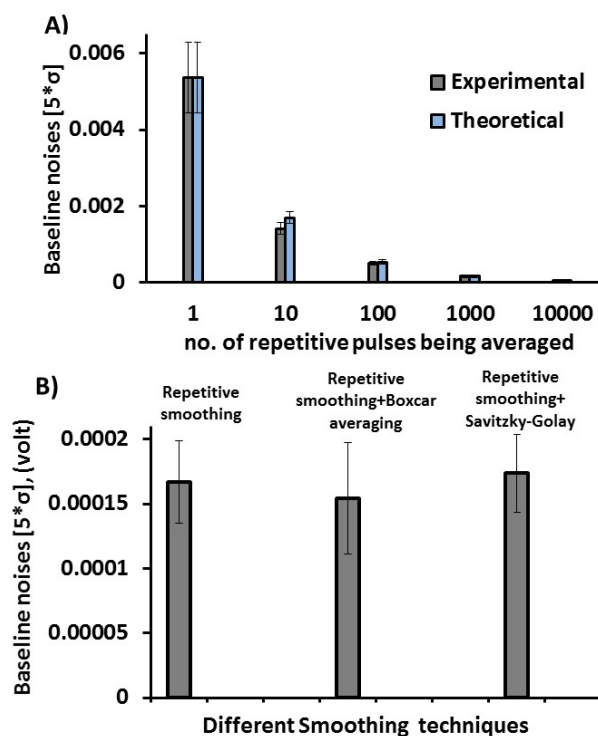


Fig S 2 A) Baseline noise values of raw and repetitive smoothing pulses compared with theoretical values B) Comparison of baseline noise values obtained after three different smoothing techniques.

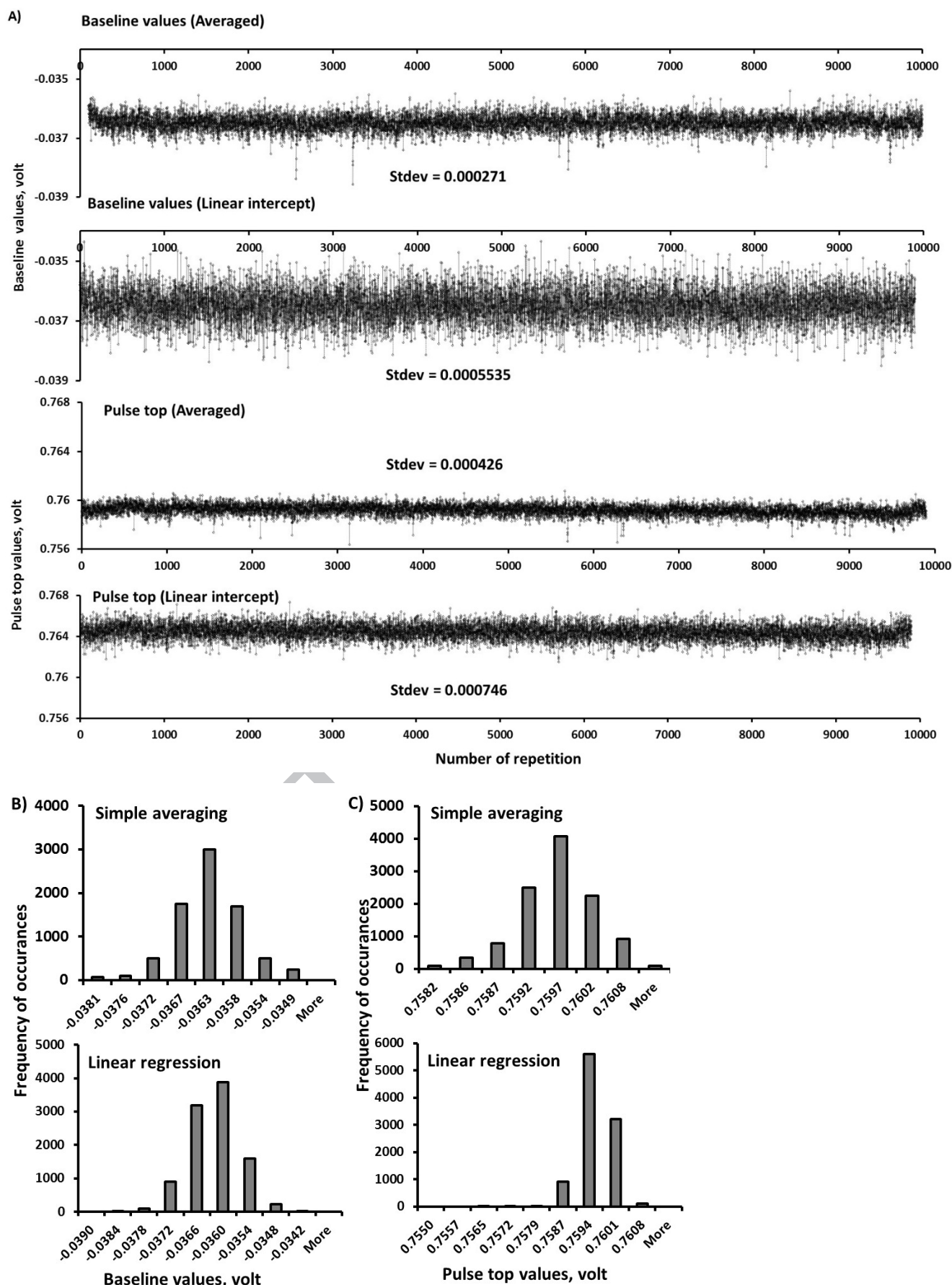


Fig S 3 A) Pulse top and baseline signal values of respective optical voltage pulses, Constructed histogram **B)** pulse top data values **C)** Baseline data values obtained from 10,000 consecutive raw pulses after applying two different data evaluation statistical methods simple averaging and linear regression.

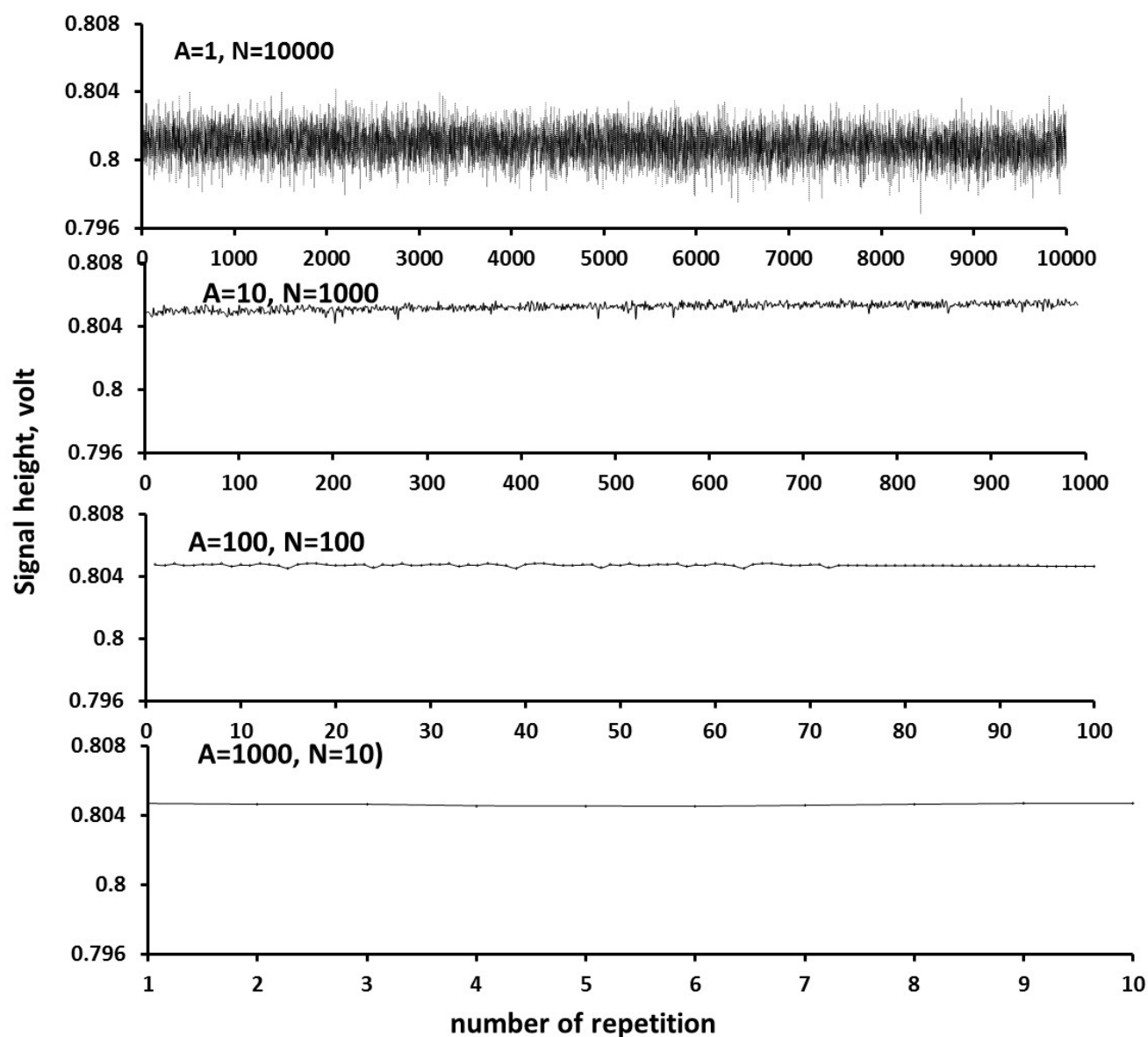


Fig. S 4 Analytical signal from each digitally processed signal pulses of processed data point as a quasi-continuous data stream using simple averaging applied on of the pulse top (125 data points) and base line (150 data points).

3. Command line parameters and code in C language

Command line parameters for the Red Pitaya peakshape program (note that lower case letters in the commands below refer to a number chosen by the user):

t,v Time (t) and voltage (v) pair. Time is in 0.01 micro seconds or steps, Voltage is in volts. Each pair specifies the next step in generating the voltage graph to be sent from the red pitaya, taking t steps to get from current voltage to desired voltage. For example if the graph is currently at 0.4V and the parameter given is 4,0.6 then the next four steps in the graph will be 0.45V 0.50V 0.55V and 0.6V.

N=n Specifies how many samples to process. The pump will be turned on before each sample. The value of n must be a whole number greater than or equal to 1. A value less than 1 will be replaced by 1. If it is not specified the default is 10. If N=n is specified more than once then only the last one on the command line is used.

S=n Specifies how many sub groups of pulses are acquired for each sample. The pump will NOT be turned on between subgroups. If S=n is specified more than once then only the last one on the command line is used. If not specified, the default is 1, ie not sub group analysis.

A=n Specifies how many pulse are generated and acquired for each sub group which are averaged together. The total number of pulses generated / acquired and then averaged together for each sample is value of S x value of A. If A=n is specified more than once then only the last one on the command line is used.

P:s,t Specifies the speed (s) and time (t) that the pump will operate between the analysis of each sample. The value of s must be between 0.5 and 1.8, if the value is less than 0.5 the pump will not turn on, if the value is greater than 1.8 the pump will turn on to 1.8. The second parameter is the time in milliseconds to turn the pump on. Example if P:0.9,4000 is specified the pump will be provided with 0.9 volts for 4 seconds between each sample. The pump cannot be turned on for less than 500 milliseconds and any value less than 500 will be taken as 500. Example if P:1.1,0 is specified the pump will turn on the 500 milliseconds. You cannot turn the pump off by specifying no time, you must specify no voltage or not include the parameter. If this parameter is not present, the default is not to turn the pump on. . If P:s,t is specified more than once then only the last one on the command line is used.

R=Y Record the time when the pump is turned on and also turned off.

C:s,f Specifies a calculation zone within the recorded averaged pulse. The calculation performed and a simple average, linear regression, polynomial regression, boxcar smoothing with simple average, linear regression and polynomial regression along with s-k smoothing with simple average, linear regression and polynomial regression. Up to 10 calculation zones can be specified.

B:n Specifies how many points to average together in the box car analysis.

V:s,f,v Not yet implemented.

D=N Specifies to only read and process fast analogue input channel 1 rather than both channel 1 and channel 2.

F=Y Specifies that the input signals should be collected at 125 MHz instead of the default 15.625 MHz.

Z=N Specifies to not zero the calculation array between samples. This parameter should rarely be used if ever.

O=N Specifies that the sequence of averaged pulses should NOT be written to the output.

O=n Specifies how many steps of each averaged pulse should be output, default if not specified is 1000. If O=n is specified more than once then only the last one on the command line is used.

T=Y Specifies that timing information should be provided to the user on stderr. The information produced is how long is spent in each routine on every call, which can be used to improve the program performance or determine how long a particular run is going to take. This parameter should rarely be used.

W=Y Indicates that the program should wait for a signal (button being pushed) before collecting and analysing each sample.

W=S Indicates that the program should initially wait for a signal (button being pushed) before starting to collect samples.

4. Program flow chart

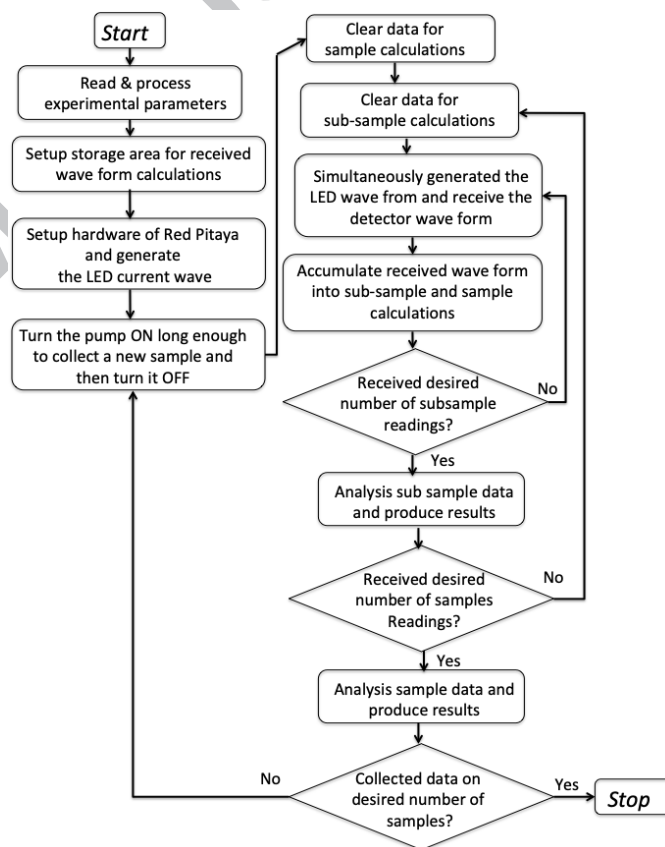


Fig. S 5 Program flowchart of the corresponding algorithm.

5. Program code: Red Pitaya function for generating LED pulse and acquiring signal from detector

```

int GenerateAcquire ( int    DualChannel,
                     int    FastRate,
                     float  in1[ ],
                     float  in2[ ] )
{
    /* This routine generates the output wave and acquires the responses.
    /* Prerequisite: The wave form has been loaded into the FPGA using
    /* rp_GenArbWaveform and the frequency set with rp_GenFreq.
    /* The array in1[ ] will be filled with the detected wave form.

    uint32_t buff_size = MaxCalculation ;
    int      TriggerWait ;

    if ( FastRate )
    { rp_AcqSetDecimation( RP_DEC_1 ) ; }
    else
    { rp_AcqSetDecimation( RP_DEC_8 ) ; }
    rp_AcqSetTriggerLevel( 0 ) ;
    rp_AcqSetTriggerDelay( 8192 ) ;
    rp_AcqStart( ) ;
    rp_AcqSetTriggerSrc( RP_TRIG_SRC_EXT_PE ) ;

    rp_GenOutEnable( RP_CH_1 ) ;
    rp_GenTriggerSource( RP_CH_1, RP_GEN_TRIG_SRC_EXT_PE ) ;

    rp_acq_trig_state_t state = RP_TRIG_STATE_WAITING ;
    TriggerWait = 0 ;
    while ( ( state != RP_TRIG_STATE_TRIGGERED ) && ( TriggerWait < 20 ) )
    { rp_DpinSetState( RP_DIO1_P, RP_HIGH ) ;
      usleep ( 5 ) ;
      rp_DpinSetState( RP_DIO1_P, RP_LOW ) ;
      usleep ( 5 ) ;
      TriggerWait++ ;
      rp_AcqGetTriggerState( &state ) ;
    }

    if ( state != RP_TRIG_STATE_TRIGGERED )
    { fprintf( stderr, "***** Trigger failed : state is %d\n", state ) ;
      return( false ) ;
    }
    else
    { usleep( 15 ) ;
      rp_AcqGetOldestDataV( RP_CH_1, &buff_size, in1 ) ;
      if ( DualChannel )
      { rp_AcqGetOldestDataV( RP_CH_2, &buff_size, in2 ) ; }
      rp_GenOutDisable( RP_CH_1 ) ;
      return( true ) ;
    }
}

```

Full detailed program code which is multiple pages long is available from the authors upon request.

Research Highlights

- Flexible data processing with field programmable gate array (FPGA) incorporated with portable open source microcontroller
- Rapid pulse generation of 2 μ s short pulses with a typical repetition rate of 1 kHz to drive the IR LED
- Optical sensing, data acquisition and processing on-the-fly of 2 Gbit/s datastream
- Continuous analytical signal of 1 point every 1 ms to 10 s obtained
- Minimum measurable absorbance of 10^{-4} a.u. achieved